

**Amendments to the Specification:**

On page 1, please replace lines 4-9 (first paragraph), with the following rewritten paragraph:

The subject matter disclosed herein is related to the subject matter disclosed in ~~eo-~~  
~~pending application Serial No. 09/417,166 (attorney docket TN146), filed October 12, 1999,~~  
U.S. Patent No. 6,823,313, November 23, 2004, "Methodology for Developing Interactive  
Systems," the contents of which are hereby incorporated by reference. In addition, we hereby  
claim the benefit of the priority date of U.S. Provisional Application ~~Serial~~ No. 60/236,360,  
filed September 28, 2000, "Dialog Flow Interpreter."

On page 2, please replace lines 26-31 (fourth paragraph), with the following rewritten paragraph:

Additional background on IVR systems can be found in U.S. Patent No. 6,094,635,  
July 25, 2000, ~~itled~~ "System and Method for Speech Enabled Application"; in U.S. Patent  
No. 5,995,918, November 30, 1999, "System and Method for Creating a Language Grammar  
using a Spreadsheet or Table Interface" and in U.S. Patent ~~Application No. 09/430,315 filed~~  
~~October 29, 1999, No. 6,510,411, January 21, 2003,~~ "Task Oriented Dialog Model, and  
Manager." ~~These patents are commonly assigned to Unisys Corporation.~~

On page 7, please replace lines 11-16 (first full paragraph) with the following rewritten paragraph:

In contrast, in accordance with the present invention, referring to Fig. 4, the developer  
would use the DFI design tool, 400, to enter a design of the whole application, as depicted in  
step 410, including many such states such as Get Social Security Number, Get PIN Number  
and so on. Once the application is rehearsed in the simulator (see ~~reference patent Serial No.~~  
~~09/417,166~~ U.S. Patent No. 6,823,313), step 420, files may be generated that represent that  
design, steps 440 and 450.

On page 11, please replace lines 3-12 (first full paragraph), with the following rewritten paragraph:

Although they address similar problems, the DFI is very different from [of] the Speech Objects model. Speech Objects set up defaults a program can override (the program has to know this from somewhere) whereas DFI provides the application with what to do next. Speech Objects are rigid and preprogrammed and of limited scope, whereas the DFI is built for a whole application and is dynamic. Speech Objects are “tuned” for a special purpose. ~~The aforementioned~~ This tuning may be provided through the DFI design tool, as well. Another way to think of the difference is that the DFI delivers “custom” speech capabilities built through the tool, including how they “link” together. Speech Objects provide “prepackaged” capabilities (with the advantage of “expert design” and tuning) and with no “flow” between them.